Federal PKI Directory Profile
1/25/2001

1. Introduction

This profile defines the requirements for the initial operational Federal Public Key Infrastructure (FPKI) directory system. The FPKI builds upon the Federal Bridge Certification Authority (FBCA) prototype that was successfully demonstrated during the Electronic Messaging Association (EMA) Challenge in April 2000. This prototype supported S/MIME messaging among several disparate PKI domains using several different CA products, X.500 directory products, and S/MIME e-mail clients. This demonstration illustrated interoperability on several levels – between CAs, between directories, and between e-mail clients. Each client created, and then processed a certificate trust path between the domain of the recipient and the domain of the sender in order to validate the signer's digital signature on the e-mail. Trust paths up to seven certificates were constructed and validated. Directories were chained using the X.500 Directory System Protocol (DSP), while the Lightweight Directory Access Protocol (LDAP) was employed by the e-mail client to access its local directory [1].

The FPKI will use a Federal Bridge CA that cross-certifies with agency Principal CAs to provide trust paths between the agencies. An FBCA directory server will be chained to agency border directories to make certificates available for PKI users. The Border CA concept is described in [2].

In the following sections, this profile will address the minimum required schema, the naming conventions, the directory protocols to support, alternatives to consider, and issues to bear in mind in order to adapt to this evolving technology. Familiarity with the PKI technology, concepts and general terms of the directory service is assumed.

The draft is based on several sections of the following documents:
- *The Evolving Federal Public Key Infrastructure* [1],
- *Governmentwide Directory Support 2 Technical Series, the Updated US Gold Schema document* [3],
- *The Bridge CA Demonstration Repository Requirements Draft 4/8/1999* [4], and
- *NSA Bridge Certification Authority Demonstration Phase II - Directory Requirements and Architecture, 7/3/2000* [5].

2. Schema Requirements

This section addresses the minimum schema requirements for agency directories to interoperate with the FPKI directory. The schema is limited to just the objects needed to support the PKI. At a minimum, the directories are required to store and disseminate the following PKI related attributes:

- Certification Authority Certificates
- Certificate Revocation Lists
- Authority Revocation Lists
- Cross Certificates
- End-entity certificates
- RFC822MailUser

In the Internet X.509v3 Public Key Infrastructure LDAPv2 Schema [6], these attributes are:
- cACertificate

- certificateRevocationList
- authorityRevocationList
- crossCertificatePair
- userCertificate
- rfc822Mailbox

This schema is used in some commercial CA products.

Some agencies may wish to make other information available externally to support their PKI applications. However, this profile does not address or impose requirements on application-specific data in agency directories.

The cACertificate and crossCertificatePair attributes require special attention when accessing the directory to build the certificate path. Neither the PKIX specification nor the X.509 standards explicitly provide an algorithm to construct a certificate path. The PKIX LDAP-V2-schema provides guidance on what can be stored in the specific attributes. The draft states the following about the cACertificate attribute and the crossCertificatePair attribute:

> The cACertificate attribute of a CA's directory entry shall be used to store self-issued certificates (if any) and certificates issued to this CA by CAs in the same realm as this CA.
>
> The forward elements of the crossCertificatePair attribute of a CA's directory entry shall be used to store all, except self-issued certificates issued to this CA. Optionally, the reverse elements of the crossCertificatePair attribute of a CA's directory entry may contain a subset of certificates issued by this CA to other CAs. When both the forward and the reverse elements are present in a single attribute value, issuer name in one certificate shall match the subject name in the other and vice versa, and the subject public key in one certificate shall be capable of verifying the digital signature on the other certificate and vice versa.
>
> When a reverse element is present, the forward element value and the reverse element value need not be stored in the same attribute value; in other words, they can be stored in either a single attribute value or two attribute values.
>
> In the case of V3 certificates, none of the above CA certificates shall include a basicConstraints extension with the cA value set to FALSE.

A path development algorithm must consider that the CA's certificate must be stored in the crossCertificatePair attribute, but the algorithm may consult the cACertificate attribute first, for performance reasons.

The following sections define the attributes and object classes that are required for end entities and CAs.

2.1 End Entities

Attributes

End entity (EE) directory entries shall contain, as a minimum, the following attributes:

1. ***userCertificate*** as defined in 1997 X.509v3 [7] (OID: 2.5.4.36),

2. ***attributeCertificate*** as defined in 1997 X.509v3 (OID: 2.5.4.58),
3. ***commonName*** as defined in 1997 X.521 [8] (OID: 2.5.4.3),
4. ***surname*** as defined in 1997 X.521 (OID: 2.5.4.4).

NOTE: The EE relative distinguished name (RDN) shall consist of the ***commonName*** attribute type and value. For example: cn=John Smith

Object Classes

EE entries shall be made up of the following object classes:

1. ***person*** as defined in 1997 X.521 (OID: 2.5.6.6).

2. ***pkiUser*** as defined in RFC 2587: LDAPv2 Schema (OID: 2.5.6.21) for non-Entrust EEs -- OR -- ***entrustUser*** as defined in "Entrust Directory Schema Requirements" version 1.0, dated August, 1998 (OID: 1.2.840.113533.7.67.0) for Entrust EEs.

3. ***securePkiUser*** as defined in ACP 133 Edition B [9] (OID: 2.16.840.1.101.2.2.3.66). This auxiliary object class includes ***attributeCertificate*** and ***supportedAlgorithms*** as optional attribute types.

Optionally, EEs may include the following object classes:

1. ***organizationalPerson*** as defined in 1997 X.521 (OID: 2.5.6.7),
2. ***inetOrgPerson*** as defined in IETF RFC 2798 [10] (OID: 2.16.840.1.113730.3.2.2).

2.2 Certification Authorities

Attributes

CA (including PCAs and PAAs) entries in the directory shall contain at a minimum the following attributes:

1. ***commonName*** OR ***organizationalUnitName*** as defined in 1997 X.509v3 (OIDs: 2.5.4.3 and 2.5.4.11 respectively).

2. ***cACertificate*** as defined in 1997 X.509v3 (OID: 2.5.4.37). As per the LDAPv2 Schema (RFC 2587), the cACertificate attribute shall be populated as follows:

   "The *cACertificate* attribute of a CA's directory entry shall be used to store self-issued certificates (if any) and certificates issued to this CA by CAs in the same realm as this CA."

3. ***certificateRevocationList*** as defined in 1997 X.509v3 (OID: 2.5.4.39)

4. ***crossCertificatePair*** as defined in 1997 X.509v3 (OID: 2.5.4.40). As per the LDAPv2 Schema (RFC 2587), the crossCertificatePair shall be populated as follows:

   "The forward elements of the crossCertificatePair attribute of a CA's directory entry shall be used to store all, except self-issued certificates issued to this CA. Optionally, the reverse elements of the crossCertificatePair attribute of a CA's directory entry may contain a subset of certificates issued by this CA to other CAs. When both the forward

and the reverse elements are present in a single attribute value, issuer name in one certificate shall match the subject name in the other and vice versa, and the subject public key in one certificate shall be capable of verifying the digital signature on the other certificate and vice versa.

"When a reverse element is present, the forward element value and the reverse element value need not be stored in the same attribute value; in other words, they can be stored in either a single attribute value or two attribute values."

CAs entries in the directory may optionally contain the ***authorityRevocationList*** attribute as defined in 1997 X.509v3 (OID: 2.5.4.38).

NOTE: The CA relative distinguished name (RDN) shall consist of either the ***commonName*** attribute type and value or the ***organizationalUnitName*** attribute type and value. For example: cn=NSA CA -- OR -- ou=ECA1

Object Classes

CA entries shall be made up of the following object classes:

pkiCA as defined in RFC 2587: LDAPv2 Schema (OID: 2.5.6.22) for non-Entrust CAs -- OR -- ***entrustCA*** as defined in "Entrust Directory Schema Requirements" version 1.0, dated August, 1998 (OID: 1.2.840.113533.7.67.1) for Entrust CAs.

The base object class of CAs shall be one (or more) of the following:

1. ***person*** as defined in 1997 X.521 (OID: 2.5.6.6)
2. ***organizationalPerson*** as defined in 1997 X.521 (OID: 2.5.6.7)
3. ***inetOrgPerson*** as defined in IETF RFC 2798 (OID: 2.16.840.1.113730.3.2.2)
4. ***organizationalUnit*** as defined in 1997 X.521 (OID: 2.5.6.5)

3. Namespace Control and DIT Structure

PKI objects are defined in the X.509 specification, and use the X.500 information model. This model is used by both X.500 and LDAP-based directory services operated by government agencies that will rely upon the Federal Bridge CA. A PKI object (such as a public key certificate or certificate revocation list) is located using that object's Distinguished Name (DN), which specifies the location of the object within the federal directory information tree. This "tree" is a logical hierarchical structure composed of all the various agency directory services. A "namespace" is the section, or subtree, of the directory controlled by a specific agency.

3.1 Agency Directory Service Requirements

Agencies are not required to conform to any specific directory protocol internally. But, in order to interoperate with the FBCA, an agency's directory service must conform to the following requirements:

- The agency's PKI information must conform to the X.500 information model and X.509.
- The agency's PKI information must conform to one of the namespace strategies stated in Sections 3.2, 3.3, and 3.4, below.

- The agency's directory service must support 1993 X.500 chained operations, 1993 X.500 referrals, or LDAP v3 referrals.

- The agency must register their directory service as in Section 3.1.1 with the FBCA in order to establish interoperability.

The agency may choose to employ a Border Directory Server Agent (DSA) to provide for protocol conversions, enforce security, and restrict access to internal directory services.

3.1.1 Registration

The FBCA will provide connectivity between, and references to, registered U.S. Government agency directory systems (see Section 3.4). In order to establish this connectivity, each agency participating in the FBCA must register their directory service or Border DSA with:

@ registration contact info goes here – name, phone #, office, etc.

The following information must be provided:

- Name and address of agency

- Name, address and contact information for agency's directory administrator

- Distinguished Name, Network Address, and Host Name of directory service

- Naming Context (namespace) kept on this directory server (see Sections 3.2, 3.3, and 3.4)

- Protocols (X.500 and/or LDAP) – at least one is required

- If X.500, state whether chained operations and/or referrals are allowed

Appendix @@ to this document contains a worksheet to aid you in collecting this information prior to registration.

3.2 X.500 Directory Services

If the agency chooses to use X.500-based directory services, its directories must conform to the name space as defined for the Federal Government [3] (Figure 3-1). This namespace contains the U.S. Government level of the global X.500 Directory Information Tree (DIT) and all governmental agencies and departments. In X.500 terms, this namespace includes directory servers with the naming context of:

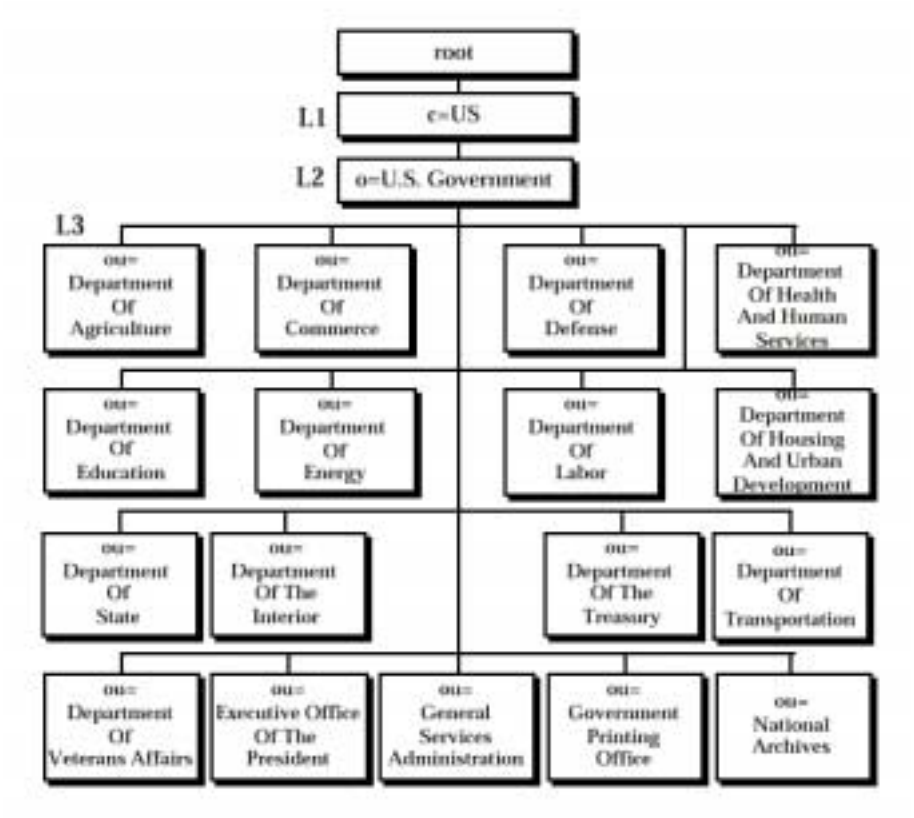
c=us; o=U.S. Government

The U.S. Government is registered as an *organization* (o) object class in the Global DIT. Agencies and departments are registered as *organizationalUnit* (ou) object classes immediately beneath the o=*U.S. Government* entry in the Global DIT. Agency and department names in the Federal Government namespace must conform to agency and department names as stated in the Federal Government Manual. This publication cites first level agencies and departments (*organizationalUnit*s) in each branch of the Federal Government. For instance, Department of Transportation would be:

c=us; o=U.S. Government; ou=Department of Transportation

The agency or department is free to define and manage the namespace for lower levels within (underneath) the agency or department level of the directory.

Figure 3-1 Federal Government Top Level DIT



Abbreviations are allowed, but must be negotiated with the FBCA Registrar (above) to ensure uniqueness within the U.S. Government namespace. Potential conflicts on abbreviations can occur, and will be solved as follows.

If the agency or department has a current domain registration within the Internet Domain Naming System (DNS) underneath *.gov*, they may use this as an abbreviation within the Federal Government directory. For example, the Department of Transportation is registered as *dot.gov*, and thus can use:

c=us; o=U.S. Government; ou=DOT

Department of the Treasury, however, is registered as *treas.gov*, and thus could use:

c=us; o=U.S. Government; ou=TREAS

Any number of organizationalUnitNames may be registered to aid in directory searches. For example, the Department of Transportation directory entry could have the following names (if authorized by the FBCA Registrar):

c=us; o=U.S. Government; ou=Department of Transportation (specified RDN)
c=us; o=U.S. Government; ou=Transportation
c=us; o=U.S. Government; ou=DOT

The actual directory entries would be found underneath the name specified in the Federal Government Manual. All other DNs would actually be aliases pointing to this "official" DN.

3.2.1 DNs versus RDNs

In the above example, a specific directory entry could be located by three different DNs. The first DN identified the actual directory entry, and the other two were aliased to the first.

Each entry contains a unique value known as the Relative Distinguished Name, or RDN. These values are chained together to form the DN for any specific object in the Global DIT. In the above examples, the DN for Department of Transportation tells us that the Directory Information Tree contains:

- A *country* object with a RDN of *c=us*
- An *organization* object with a RDN of *o=U.S. Government*, which is subordinate to the *c=us* object.
- An *organizationalUnit* object with a RDN of *ou=Department of Transportation*, which is subordinate to the *o=U.S. Government* object.

X.500 and LDAP allow for multi-value attributes. For instance, the Department of Transportation object could contain *commonName* attributes for all three names – Department of Transportation, Transportation, and DOT. This feature allows users to find the object by specifying part of the name, rather than traversing the DIT.

3.2.2 Advantages and Disadvantages of X.500

The X.500 naming scheme is well understood. It has been supported in current PKI products, which have been successfully demonstrated in the PKI BCA and the EMA challenge demonstrations. However, the drawback of this naming scheme is that it is little used by anyone other than for PKI. Generally speaking, users do not necessarily understand or care about the finer distinctions of the Federal structure, therefore, distinguished names with organizational structure embedded in them are difficult for users to comprehend or remember.

In addition, the more structure that is embedded in names, the more certificates that would need to be revoked when structures change. And the more structure that is built into the names, the more the name space needs to be administered. Many agencies have adopted a very "flat" namespace, where all the organization's users are listed directly underneath the agency object or within a single subtree, regardless of location or organizational structure.

Another recurring debate, which occurs with X.500-based systems, lies in the directory tree structure within the agency. There are three basic approaches:

- Put all the directory entries into a single, flat namespace (usually requires a single DSA serving the entire agency).

- Divide the tree to mirror organizational structure (may create problems if the directory servers are located in multiple geographic locations).

- Divide the tree to mirror geographical or network infrastructure (presents issues related to interactive searching and use).

The Federal Bridge CA has no preference and issues no guidance as to the tree structure of internal agency directory services. This area is clearly outside the scope of this document.

3.3 Internet Domain Name Based Naming

With the global acceptance of Internet and technologies such as the Domain Name System (DNS) and RFC822-based e-mail, many portions of the government have ignored older technologies such as X.500 and have implemented Internet-based infrastructures. These infrastructures are used primarily for e-mail and web-based delivery of services and information.

The Internet DNS provides a hierarchical naming and locating system based on domain name components. For instance, the Internal Revenue Service is registered as *irs.treas.gov*. The U.S. Federal government "owns" the *gov* "top-level domain", and is responsible for assigning and administering domain component names underneath that domain. Department of the Treasury (Treasury) has registered the domain component of "*treas*", underneath *gov*. Therefore, any e-mail user at the Department of the Treasury would have an e-mail address something like *any.user@treas.gov*, and the main Treasury web page would be found at *www.treas.gov*.

The Internal Revenue Service has been assigned the domain component of "*irs*" by Treasury, such that a user within IRS would have an email address of *any.user@irs.treas.gov* and the main IRS web page would be found at *www.irs.treas.gov*.

This DNS-style of naming was originally developed to support hierarchical management and searching of computer system names (e.g. "hostnames"). Each computer attached to the Internet has an Internet Protocol (IP) address, which consists of four numbers between 0 and 255, separated by periods. These addresses look something like 192.248.32.14. Clearly, this is hard for users to comprehend, much less remember. Who wants to address an email message to john.smith@192.248.32.14? (Actually, this address *will* work on many Internet-connected systems). DNS maps this numeric IP address into a human-readable system name, called a Fully Qualified Domain Name, or FQDN. This allows a user to send email to *john.smith@company.com* instead of trying to remember the IP address. The computer looks up *company.com*, finds the numeric address, and makes the connection. In this sense, IP addresses are like telephone numbers, and DNS is like a giant, worldwide electronic phone book.

X.500 is a completely separate directory system from DNS. However, a proposed Internet Standard as described in RFC 2247 [11] and RFC 2377 [12] provides a method of representing Domain Name System domain components using the X.500 information model. This allows both X.500 and LDAP-based directory services to store information in a structure familiar to Internet-literate users.

RFC 2247 defines an attribute, *DomainComponent (dc)*, which can be used to store a domain component such as "*gov*". It also defines two objects, *domain* and *dcObject*. The *dcObject* object can be added to existing objects so that they can contain a *dc* attribute. The *domain* object allows the addition of new entries that contain a *dc* attribute.

Using *domain* objects, it is possible to accurately represent the DNS "tree" within an X.500 or LDAP directory service (Figure 3-2). The user specified by the email address *john.smith@irs.treas.gov* would be represented by the X.500 DN:

dc=gov; dc=treas; dc=irs; pn=john.smith

LDAP allows a relaxed form of DN in reverse order separated by commas, which looks like:

pn=john.smith, dc=irs, dc=treas, dc=gov

The information in the directory is the same either way. Searching based on this DNS-style naming can be very intuitive to users who are familiar with Internet email addresses. The Federal Bridge CA will allow agencies to choose to implement naming in this fashion, instead of (or in addition to) the X.500-style Federal Government naming set forth in Section 3.1.

Additionally, the *dcObject* object can be used to add the *dc* attribute to other X.500 objects. Therefore it can allow for construction of DNs which look very much like X.500, but which are actually composed of *DomainComponent* attributes. This sort of DN would look like:

dc=us; dc=U.S. Government; dc=treas; dc=irs; pn=john.smith (or)
pn=john.smith, dc=irs, dc=treas, dc=U.S. Government, dc=us

The Federal Bridge CA will ***not*** support this style naming. Its similarity to pure X.500 naming would cause significant confusion. Since it doesn't map to the Internet-style e-mail addresses, it is not intuitive to use and therefore provides no benefit. As DNS evolves in the future, country-based naming may come into use. If so, this decision will be revisited at that time.

3.3.1 Drawbacks of DNS-Style Naming

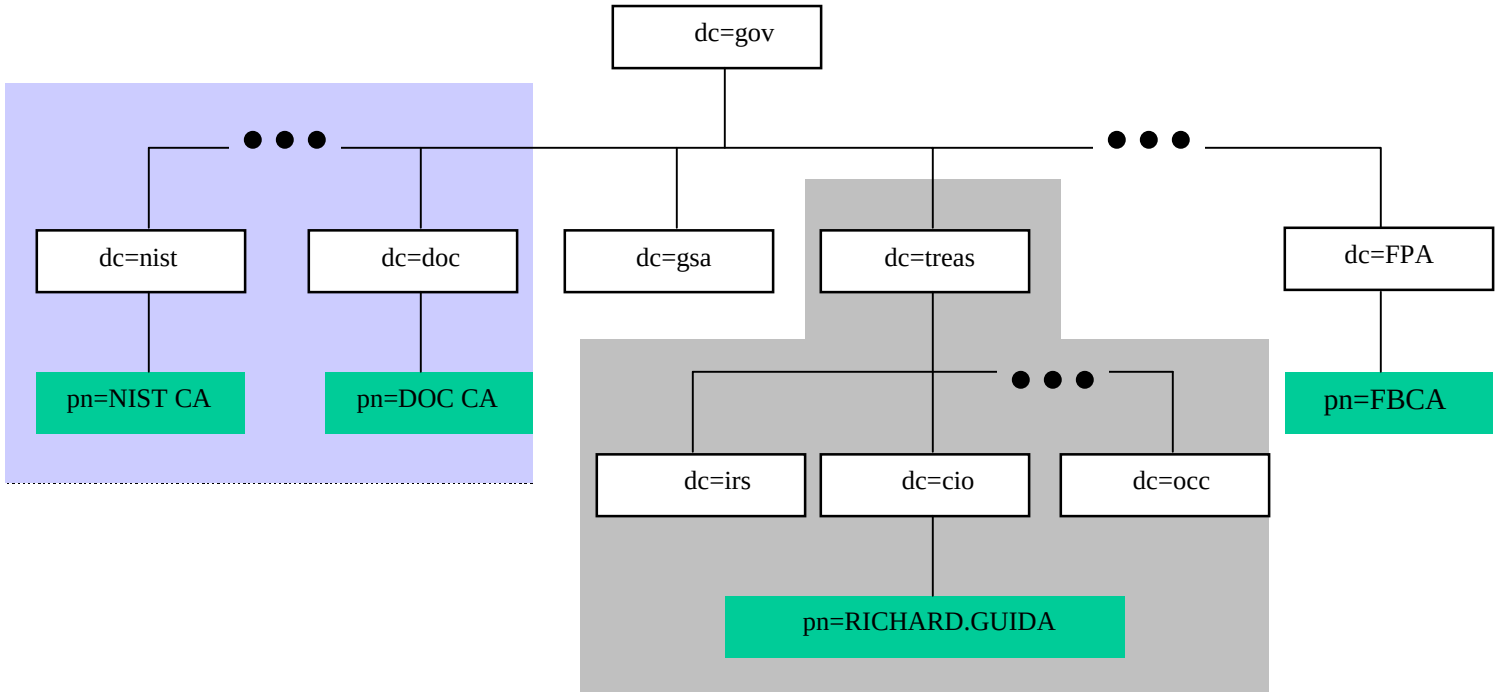
RFC 2247, the document that proposes this style of addressing, is a proposed Internet Standard. It therefore is fairly stable and not subject to major changes. However, it may not be widely implemented in applications and commercial software products yet.

The *gov* domain is owned by the U.S. Government. Registration of government agencies and operation of the government-level DNS is outsourced to a vendor. As far as is known, there is no official guidance relating to the creation of DNS-style domain information for government agencies. This leads to several confusing situations:

Internet domain name components are typically short and cryptic. Many times, all users appear directly underneath the organization with no clue as to organizational structure or geographic location. Also, many agencies have registered domain names that don't reflect the actual Federal departmental structure. This may be because of grandfathering (an agency got the name before any policy was established), or because the public is neither interested in, nor knowledgeable about the government's departmental structure, and would simply be confused by domain names that reflect actual structure. Examples include:

faa.gov	rather than	*faa.dot.gov*
nist.gov	rather than	*nist.doc.gov*
cg.mil	rather than	*cg.dot.gov*

Figure 3-2 Domain Component Naming DIT



It may be fairly clear that the FAA should be a part of the Transportation Department, but does the public generally know that NIST is a part of the Commerce Department, or that the Coast Guard, a uniformed service, is actually under the Transportation rather than the Defense Department?

Another potential problem can be confusion between the government and the private sector because of the Top Level Domain Names. The U.S. Government only has authority over domain names ending with *.gov*. Sites such as *www.irs.com, www.fbi.com,* and most notoriously (don't go there) *www.whitehouse.com* play off of this confusion for purposes of social satire, political commentary, and worse.

And lastly, there is no automatic synchronization between X.500 and the DNS. When a domain component is registered in the DNS, it will require a second action to have it manually entered into the X.500 directory. This presents the potential for the X.500 or LDAP-based directory to get out of synchronization with the current state of the DNS. Within government, the changes are infrequent enough that this may be a manageable problem.

3.4 Combined Domain Component Names with X.500 Names

Recently the Higher Education community, in a part of the Higher Ed, Internet II effort [13], has taken a slightly different approach to the use of domain component names, and asked the FPKI directory profile support this option. This community advocates combining domain component names with traditional X.500 names in the subjectName field of a certificate to enforce name uniqueness. This requires no new registration or management, and it may faciliate directory service discovery via DNS SRV records [14]. No rule in X.500 prohibits this, recent changes to the FBCA CP will also allow for this flexibility. New infrastructures are being designed in the Internet2/EDUCAUSE arenas to meet the needs of academia and a myriad of applications [13]. Allowing this flexibility will facilitate interoperability between institutions of higher education and the federal government, and foster the use of the FBCA model outside the US government.

The directory working group has discussed this proposal extensively and tentatively agreed to support this option as a reasonable basis for interoperable naming. The FBCA would stand up a directory server with 2 (or 3) roots for [o=US Government, c= US], [dc=gov], and, possibly, [dc=mil]. Agencies would be encouraged to include the combined name form in entity certificates and could choose whether to use [o=US Government, C= US] (Figure 3.3) or [dc=gov] (Figure 3.4) as the most significant part of their name. It would also be acceptable to use only one name form or the other (Figure 3.1 and Figure 3.2).

Using this scheme, some equivalent examples would be:

1. cn=John Smith, dc=irs, dc=treas, dc=gov, ou=Department of Treasury, o=U.S. Government, c=US

2. cn=John Smith, dc=irs, ou=Internal Revenue Service, dc=treas, dc=gov, ou=Department of Treasury, o=U.S. Government, c=US

3. cn=John Smith, ou=Internal Revenue Service, dc=irs, dc=treas, dc=gov, ou=Department of Treasury, o=U.S. Government, c=US

Or, starting with the ".gov" domain name:

1. cn=John Smith, ou=Internal Revenue Service, o=U.S. Government, dc=irs, dc=treas, dc=gov

Figure 3.3. Combined DCN with X.500 names -1

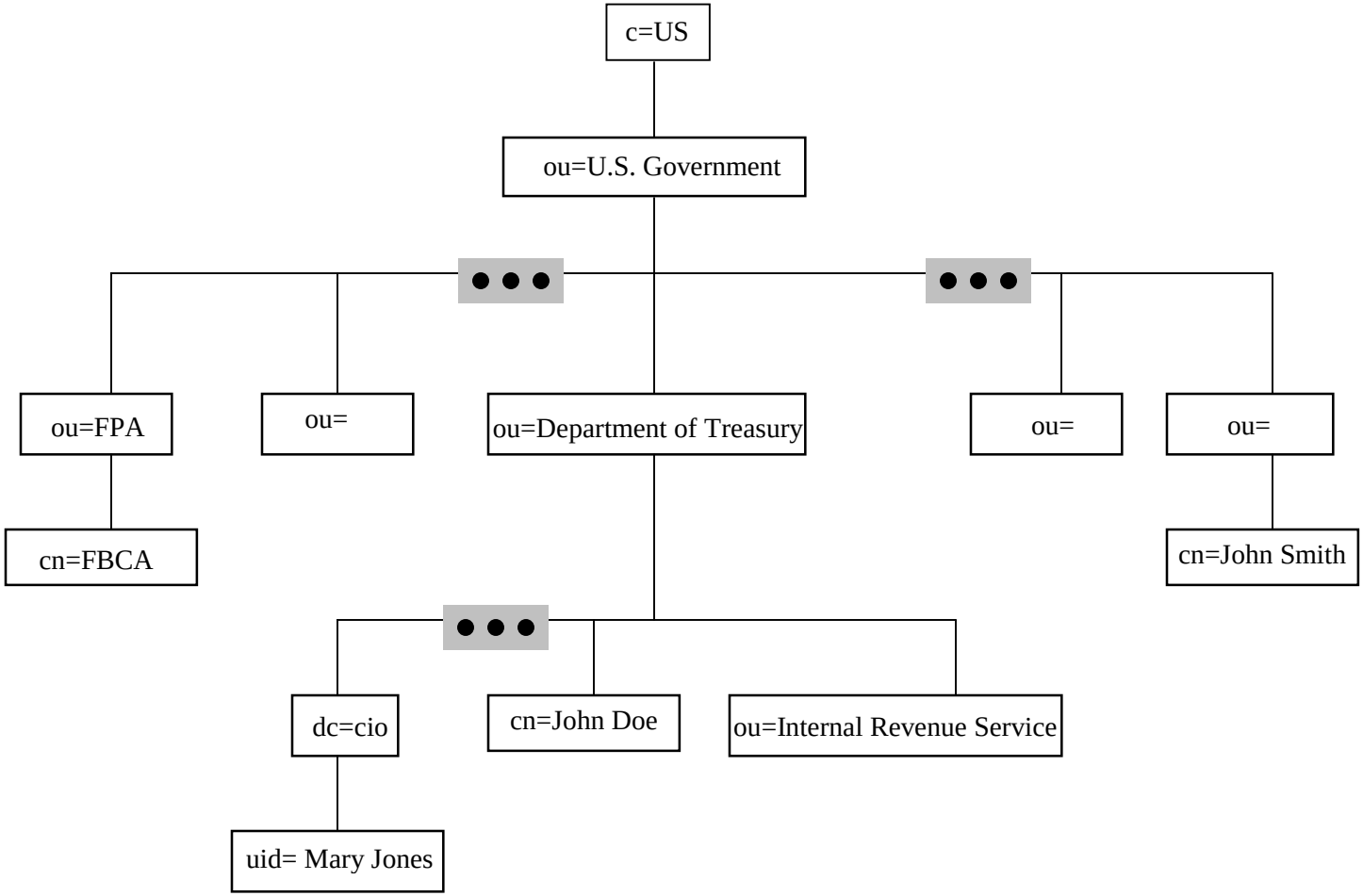
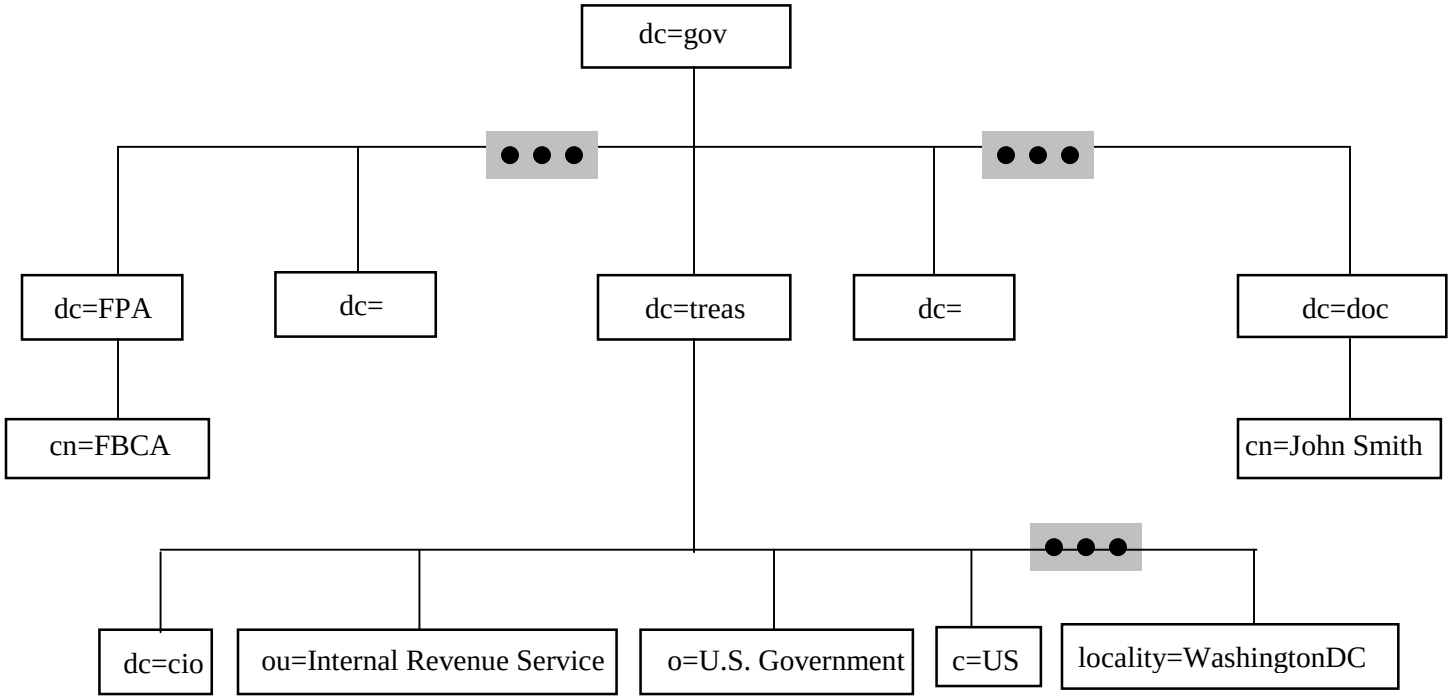


Figure 3-4 Combined DCN with X.500 Names - 2



2. cn=John Smith, ou=Internal Revenue Service, o=U.S. Government, c=US, dc=irs dc=treas, dc=gov

Several issues have been raised regarding this combined naming scheme. One is, do X.500 DSA products "object" to seeing the "c=" attribute subordinate to the "dc=". Are there other features of this naming scheme that "break" some directory products? What are the rules, if any, for formulating the combined names? For example all the names above start (on the right) with either the "c=US" or "dc=gov" attribute and end (on the left) with the common name. This makes sense intuitively, but does it make any difference to the processing of the name? These issues are yet to be explored.

3.5 The U.S. Government Directory Server

In order to promote interoperability between various agency and department directory services, the Federal Bridge CA program will operate a Directory Server that supports both the Federal Bridge CA, and the U.S. Government level of the X.500 DIT.

In support of the U.S. Government level, the FBCA program will provide the following services:

- Registration of directory services for agencies that wish to (a) participate in the Bridge CA program, and/or (b) interoperate with other government directory services.
- The DSA will provide knowledge references to all registered directory services, and also to international government and the private sector, as required in order to promote Electronic Government initiatives.
- Coordination with E-gov and international interoperability initiatives.

The DSA will support the traditional X.500 DIT for the U.S. Government (Figure 3.1), the "de-facto" Internet DNS directory structure (Figure 3.2), as well as the hybrid DITs as illustrated in Figure 3-3 and Figure 3.4. It will be able to bridge among these namespaces, promoting interoperability among agencies that have implemented traditional X.500 naming, those that rely upon the DNS structure, and those supported both.

4. Directory Protocols

Two broad categories of directory servers are currently in use: "X.500 DSAs", and "LDAP servers." Both use the same X.500 directory information model and the LDAP v2 or v3 client directory access protocol. An X.500 DSA also supports Directory Service Protocol (DSP) chaining of directory servers. An LDAP server typically supports the LDAPv3 [15] client interface and LDAPv3 referrals. If chaining between servers is offered, it is usually a proprietary implementation.

Most agencies will choose to operate with the Federal PKI through a border directory server located outside the agency firewall rather than through an internal agency directory server. However this profile does not preclude chaining internal directory servers to the FBCA directory server.

The FBCA will maintain an X.500 DSA, holding the roots for *c=US, o=U.S. Government, dc=gov,* and possibly, *dc=mil*. This FBCA DSA will be available for chaining to agency X.500 DSAs.

For agencies that use X.500 DSAs for their directory service, or their border directory, it is not necessary to specify the precise client to directory server access protocol. Typically, it will be some version of LDAP, but the older X.500 Directory Access Protocol (DAP) is also acceptable. All that is required is that agency clients are compatible with agency servers. Agency servers will obtain needed external certificates and CRLs for their clients via DSP chaining, and this is transparent to the clients. Each agency border directory will be chained to the FBCA directory, via DSP chaining.

Agencies that choose to use LDAP servers internally may make external agency certificates available to clients in several ways:

- The agency may stand up an X.500 DSA as a border directory and chain it to the FBCA DSA;

- Alternatively, if agency clients support LDAP v3 with referrals, then the LDAP servers may refer clients to the FBCA DSA for external certificates (or may make direct referrals to the border directories of other agencies).

Agencies that choose to use LDAP servers internally may make internal agency certificates and CRLs available externally by:

- Standing up an X.500 DSA chained to the FBCA DSA and posting externally available certificates and CRLs to it. This may be achieved by purchasing directory services from a 3rd party supplier. This is the preferred or recommended method of interoperating with other agencies through the FBCA DSA;

- Alternatively, if no X.500 border DSA is set up, users may include a certificate list beginning with the certificate issued by the FBCA to their agency PCA and ending with the user's signature certificate in the header of signed S/MIME messages. This does not directly support encryption, but it allows an external relying party (who interoperates through the FBCA) to validate S/MIME signatures.

As the Federal PKI develops, the FBCA directory may incorporate an meta-directory capability, to transparently resolve the queries of X.500 DSAs for information contained in LDAP servers. This capability, however, will not be a part of the initial FBCA directory.

In principle, the choice to use X.500 style or Domain component names is independent of the choice to use X.500 DSAs or LDAP servers. In practice, it appears likely that those who choose to use domain component names will probably choose to use LDAP servers. It is possible to chain through the FBCA DSA from an agency that uses Domain Component names to one that uses X.500 style names. The FBCA directory shall hold the root for both *c=US, o=U.S. Government* and *dc=gov*, and support chaining of both name types.

4.1 Authentication Requirements

Directories are required to support simple authentication for LDAP and DSP communications.

4.1.1 Client Authentication

FPKI directory clients that read the FPKI directory (read, list, search directory operations) require no authentication (i.e. anonymous bind to the directory is acceptable). This profile does not

address directory access control requirements to update FPKI directory servers. Agencies must ensure that only authorized parties can update FPKI directories.

4.1.2 Server Authentication

FPKI directories are required to support simple authentication for server to server chaining (X.518 DSP) communications.

(Colin.Robbins@nexor.com has argued for no authentication for DSP for the following reason, which will not appear in the final draft of the profile. The WG needs to decide on this issue.)

> If we are going to use simple password based authentication for DSP, then we have to define a schema for registering DSA entries, and most importantly where in the DIT these entries are named. Nearly all product vendors will want the DSA entry name in the part of the DIT their DSA manages - it makes life easy for them. Regrettably this approach breaks large-scale distributed operations.
>
> If DSA A wants to connect to DSA B, it will pass a user name and password to DSA B. DSA B needs to check this somewhere locally to verify the identity of DSA A. Where does it get this password from? The natural place is in the DSAs entry in the DIT. Alas, this is stored in DSA A, so you cannot get at it (unless all DSAs implement the complex call-back bind-compare operation chaining model).
>
> So, what often happens, is that password is configured by an administrator locally. This works for a while, but it does not scale to a distributed directory of hundreds of servers. So, to use DSP password, you have to define a mechanism for managing and distributing these passwords. The only viable I have come across way is to use the DIT itself, and ensure the part of the DIT storing the DSA entries is highly replicated. However this is generally an unpopular approach. (There was once an Internet-Draft proposing such a schema, but it never made it very far).
>
> Interestingly this problem does not exist for strong authentication. Let us also bear in mind that simple passwords offer no real security.
>
> Consequently, I propose that for DSP no authentication is used, until such time Strong Authentication is viable.

DISCLAIMER

The FBCA directory service is being provided to promote full interoperability between government agencies, in support of the Bridge CA. Every attempt will be made to ensure that information contained in the directory service is correct (as provided by the individual agencies), and that this information is protected from unauthorized access and modification. However, each agency or department must consider the possible consequences of unintended disclosure of information provided due to error or attack. It is the responsibility of each agency or department to establish their own policy and security posture with regard to directory-based information, and to implement whatever protocols and protection that they deem sufficient to protect critical systems, including their internal directory services.

References

[1] The Evolving Federal Public Key Infrastructure, Federal Public Key Infrastructure Steering Committee, Federal Chief Information Officers Council, gits-sec.treas.gov.

[2] Burr, W., "Public Key Infrastructure (PKI) Technical Specifications: Part A-Technical Concept of Operations", September 1998

[3] Governmentwide Directory Support 2 Technical Series, the Updated US Gold Schema document, 7/14/1997, by Booz Allen & Hamilton.

[4] The Bridge CA Demonstration Repository Requirements Draft 4/8/1999 by Chromatix, Inc.

[5] NSA Bridge Certification Authority Demonstration Phase II - Directory Requirements and architecture, 7/3/2000, by Entigrity Solutions.

[6] Boeyen, S., Howes, T., and P. Richard, "Internet X.509 Public Key Infrastructure LDAPv2 Schema", RFC2587, June 1999.

[7] ITU-T Recommendation X.509 (1997) | ISO/IEC 9594-8: 1997, “Information technology - Open Systems Interconnection - The Directory: Authentication framework”, June 1997.

[8] ITU-T Recommendation X.521 (1997) | ISO/IEC 9594-7: 1997, “Information technology - Open Systems Interconnection - The Directory: Selected object classes”.

[9] Common Directory Services and Procedures, ACP (Allied Communication Publication) 133 Edition B, March 2000.

[10] M. Smith, "Definition of the inetOrgPerson LDAP Object Class", RFC 2798, April 2000.

[11] Kille, S., Wahl, M., Grimstad, A., Huber, R., and S. Sataluri, "Using Domains in LDAP Distinguished Names", RFC 2247, January 1998.

[12] Grimstad, A., Sataluri, S., and M. Wahl, "Naming Plan for Internet Directory-Enabled Applications", RFC 2377, September 1998.

[13] The Middleware Architecture Committee for Education (MACE) http://middleware.internet2.edu/MACE/

[14] Armijo, M., Leach, P., Esibov, L., and R. Morgan, “Discovering LDAP Services with DNS”, Internet Draft <draft-ietf-ldapext-locate-o4.txt>, August 2000.

[15] Wahl, M., Howes, T., and S. Kille, "Lightweight Directory Access Protocol (v3)", RFC 2251, December 1997.